Amendment to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of the Claims:

1-2 (Canceled)

3. (Currently Amended) A ~~Java computing environment~~ **method** as recited in claim [[2]] **10**,

~~wherein said Java heap includes a plurality of Java heap portions, and~~

wherein each of said plurality of Java heap portions is designated for storing Java objects with similar traits.

4. (Currently Amended) A ~~Java computing environment~~ **method** as recited in claim [[3]] **10**, wherein each of said plurality of Java heap portions is designated to store objects of the same size.

5. (Currently Amended) A ~~Java computing environment~~ **method** as recited in claim [[2]] **10**, wherein said ~~at least one~~ **first designated** Java heap portion is implemented as an array.

6. (Currently Amended) A ~~Java computing environment~~ **method** as recited in claim [[2]] **10**, wherein said ~~at least one~~ **first designated** Java heap portion is designated to store objects with similar life spans.

7. (Canceled)

8. (Currently Amended) A Java computing environment as recited in claim [[2]] **10**, wherein said ~~at least one~~ **first designated** Java heap portion is designated to store objects that do not reference other objects.

9. (Canceled)

10. **(Currently Amended)** A method for creating Java objects in a heap **memory which is used by a virtual machine**, said method comprising:

generating, by said virtual machine, a plurality of designated Java heap portions which are each designated for storing objects that have similar class-traits, wherein said plurality of designated Java heaps include a first designated heap portion that is designated to store objects associated with a first class name, and a second designated heap portion that is designated to store objects associated with a second class name, and wherein said first and second class names are different;

receiving, by said virtual machine, a Java Bytecode that represents an instruction for creating a first Java object of said first class name;

translating [[a]] **, by said virtual machine, said** Java Bytecode into one or more commands, ~~said Java Bytecode representing an instruction for creating a Java object, and~~ wherein said one or more commands can operate to allocate said **first** Java object in [[a]] **said first designated heap** portion ~~of heap designated for said object~~; and

receiving, by said virtual machine, another Java Bytecode representing an instruction for creating a second Java object of said second class name;

translating, by said virtual machine, said other Java Bytecode into one or more other commands, and wherein said one or more other commands can operate to allocate said second Java object in said second designated heap portion;

executing**, by said virtual machine,** said one or more commands to create said **first** object in said **first designated heap** portion ~~of said heap associated with said object~~; **and**

executing said other one or more commands to create said second object in said second designated heap portion.

11. (Original) A method as recited in claim 10, wherein said method further comprises:

marking a Java Bytecode representing an instruction for creating a Java object.

12. (Original) A method as recited in claim 11,

wherein said marking is performed at compile time; and

wherein said one or more commands are created at compile time.

13. **(Currently Amended)** A method as recited in claim 10,

wherein said **first designated** portion of the heap is designated for allocation of objects with similar traits; and

wherein objects that do not have similar traits as said object**s** are not allocated in said **first designated** portion of the heap.

14-26. (Canceled)

27. (New) A computer system for creating Java objects in a heap memory which is used by a virtual machine, comprising:

at least one processor;

at least one heap memory; and

a virtual machine which is capable of:

generating a plurality of designated Java heap portions which are each designated for storing objects that have similar class-traits, wherein said plurality of designated Java heaps include a first designated heap portion that is designated to store objects associated with a first class name, and a second designated heap portion that is designated to store objects associated with a second class name, and wherein said first and second class names are different;

receiving a Java Bytecode that represents an instruction for creating a first Java object of said first class name;

translating said Java Bytecode into one or more commands, wherein said one or more commands can operate to allocate said first Java object in said first designated heap portion; and

receiving, another Java Bytecode representing an instruction for creating a second Java object of said second class name;

translating said other Java Bytecode into one or more other commands, and wherein said one or more other commands can operate to allocate said second Java object in said second designated heap portion;

executing said one or more commands to create said first object in said first designated heap portion; and

executing said other one or more commands to create said second object in said second designated heap portion.

28. (New) A computer system as recited in claim 27, wherein each of said plurality of Java heap portions is designated for storing Java objects with similar traits.

29. (New) A computer system as recited in claim 27, wherein each of said plurality of Java heap portions is designated to store objects of the same size.

30. (New) A computer system as recited in claim 27, first designated Java heap portion is implemented as an array.

31. (New) A computer system as recited in claim 27, wherein said first designated Java heap portion is designated to store objects with similar life spans.

32. (New) A computer system as recited in claim 27, wherein said first designated Java heap portion is designated to store objects that do not reference other objects.

33. (New) A computer readable medium including at least computer program code for creating Java objects in a heap memory which is used by a virtual machine, comprising:

computer program code for generating, by said virtual machine, a plurality of designated Java heap portions which are each designated for storing objects that have similar class-traits, wherein said plurality of designated Java heaps include a first designated heap portion that is designated to store objects associated with a first class name, and a second designated heap portion that is designated to store objects associated with a second class name, and wherein said first and second class names are different;

computer program code for receiving, by said virtual machine, a Java Bytecode that represents an instruction for creating a first Java object of said first class name;

computer program code for translating, by said virtual machine, said Java Bytecode into one or more commands, wherein said one or more commands can operate to allocate said first Java object in said first designated heap portion; and

computer program code for receiving, , by said virtual machine, another Java Bytecode representing an instruction for creating a second Java object of said second class name;

computer program code for translating, by said virtual machine, said other Java Bytecode into one or more other commands, and wherein said one or more other commands can operate to allocate said second Java object in said second designated heap portion;

computer program code for executing, by said virtual machine, said one or more commands to create said first object in said first designated heap portion; and

computer program code for executing, by said virtual machine, said other one or more commands to create said second object in said second designated heap portion.

34. (New) A computer readable medium as recited in claim 33, wherein each of said plurality of Java heap portions is designated for storing Java objects with similar traits.

35. (New) A computer readable medium as recited in claim 33, wherein each of said plurality of Java heap portions is designated to store objects of the same size.

36. (New) A computer readable medium as recited in claim 33, first designated Java heap portion is implemented as an array.

37. (New) A computer readable medium as recited in claim 33, wherein said first designated Java heap portion is designated to store objects with similar life spans.

38. (New) A computer readable medium as recited in claim 33, wherein said first designated Java heap portion is designated to store objects that do not reference other objects.